A New Passive 3D Automatic Target Recognition Architecture for Aerial Platforms

O. Kechagias-Stamatis and N. Aouf

***Abstract*—Three-dimensional automatic target recognition (ATR) has many advantages over its 2D counterpart, but there are several constraints in the context of small, low-cost unmanned aerial vehicles (UAVs). These limitations include the requirement for active rather than passive monitoring, high equipment costs, sensor packaging size, and processing burden. We therefore propose a new structure from motion (SfM) 3D ATR architecture that exploits the UAV's onboard sensors, i.e. the visual band camera, gyroscope and accelerometer, and meets the requirements of a small UAV system. We tested the proposed 3D SfM ATR using simulated UAV reconnaissance scenarios and found that the performance was better than classic 3D light detection and ranging (LIDAR) ATR, combining the advantages of 3D LIDAR ATR and passive 2D ATR. The main advantages of the proposed architecture include the rapid processing, target pose invariance, small template size, passive scene sensing, and inexpensive equipment. We implemented the SfM module under two keypoint detection, description and matching schemes, with the 3D ATR module exploiting several current techniques. By comparing SfM 3D ATR, 3D LIDAR ATR and 2D ATR, we confirmed the superior performance of our new architecture.**

***Index Terms*—3D Automatic Target recognition, Passive Target Recognition, Structure from Motion, Unmanned Aerial Vehicles**

I. Introduction

AUTOMATIC target recognition (ATR) is an active research field for military applications because it can enhance the quality of reconnaissance in a hostile environment. Current research involves both 2D and 3D data, including solutions based on 2D infrared (IR) [1,2], 2D synthetic aperture radar (SAR) [3,4] or inverse SAR (ISAR) [5], 2D hyper-spectral imagery [6] and 3D light detection and ranging (LIDAR) [7–11], the latter also including laser-induced fluorescence spectroscopy [12]. The military applications of ATR in several data domains have been reviewed [13].

LIDAR-based 3D target recognition is superior to its 2D counterpart because 3D encoding can exploit the geometric properties and underlying structure of an object, offering more information than 2D encoding. Indeed, features extracted from the 3D domain are affected to a lesser extent by illumination variation and target pose changes [9,14] and they can operate well in the context of a single 3D model template [10,11]. Despite these advantages, ATR based on 3D LIDAR also has several drawbacks when used with small, low-cost, time-critical unmanned aerial vehicles (UAVs) such as the RQ-11 Raven, including the disproportionate hardware cost of a LIDAR device, its large size and power requirements, the low data acquisition rate, and most importantly, the computational resources required to manipulate 3D data. For military applications, LIDAR is an active device which therefore reveals the UAV's location. In contrast, the advantages of 2D ATR include the small and inexpensive equipment, short processing times, and limited power requirements.

Here we propose an architecture that combines the advantages of 3D and 2D ATR by exploiting a structure from motion (SfM) 3D reconstruction concept that relies on a single visual band camera placed on a flying UAV platform. This is important because we demonstrate that SfM 3D ATR preserves the capabilities of 3D ATR, such as pose and illumination invariance, revealing the underlying structure of the target and relying on a single template. But SfM 3D ATR also retains the benefits of 2D ATR, such as the low processing burden, inexpensive hardware (camera *vs* LIDAR), faster data acquisition rate, and passive monitoring, the latter rendering it undetectable (Table I).

In the context of SfM based 3D ATR, current literature suggests a semantic SfM has been proposed, which simultaneously considers the geometric and semantic cues provided by 2D images [15]. However, the processing burden is 20 min per scene, making it unsuitable for UAV applications that require near-real-time processing. Brostow *et al.* [16] have demonstrated the capabilities of object recognition using an SfM point cloud, albeit with simple objects involving non-real-time 3D reconstruction. Liebe *et al.* [17] propose SfM object recognition based on 2D rather than 3D data, thus preserving the constraints of 2D ATR [17]. The usefulness of SfM has been demonstrated in military applications but only preliminary aspects of ATR were addressed [18]. Indeed, the applications of SfM have largely focused on slow-moving ground platforms rather than ATR [19], although one exceptional case (not extended to ATR) involved drone navigation [20]. Ultimately, SfM-based 3D ATR has not yet received sufficient attention, a challenge we address by proposing an innovative architecture.

The rest of the paper is organized as follows. Section II of this paper introduces the SfM 3D ATR architecture, and then Section III evaluates our method by testing it against highly credible simulated scenarios, challenging a number of current 3D ATR descriptors. The contents of the paper are

The authors are with the Signals and Autonomy group, Centre for Electronic Warfare Information and Cyber, Cranfield University Defence and security, Shrivenham, SN6 8LA, UK (o.kechagiasstamatis @cranfield.ac.uk)

TABLE I
ANALYSIS OF DATA DOMAIN DRIVEN ATR SOLUTIONS

	3D LIDAR	3D SfM	2D
Penetration of sparse structures	+	-	-
Template size	+ (can use one 3D model)	+ (can use one 3D model)	- (multiple views)
Target pose invariance	+	+	-
Target illumination invariance	+	+	-
ATR based on underlying structure	+	+	-
ATR based on texture	+ (during keypoint description)	+ (during keypoint description)	+
Operating day and night	+	-	-
Processing time	high	low	low
Equipment cost	-	+	+
Equipment size	medium / large	very small	very small
Power consumption	medium	very small	very small
Data acquiring rate	- (scanning LIDAR) + (flash LIDAR)	+	+
Maximum operating range	≈100m	>100m	>100m
Reveal sensor position	Yes (active)	No (passive)	No (passive)

summarized in Section IV.

II. SFM 3D ATR ARCHITECTURE

The proposed ATR architecture extends a previously suggested pipeline [10] to generate and utilize a 3D SfM-based point cloud. The architecture comprises offline and online phases.

A. Offline phase

During the offline phase, we use the hidden point removal (HPR) algorithm [21] to simulate an aerial view $\boldsymbol{P_m}$ of the target's computer-aided design (CAD) model as a template. $\boldsymbol{P_m}$ is then uniformly subsampled at 0.3-m resolution and described using one of the 3D descriptors presented in Section III-B.

B. Online phase

This phase comprises the SfM module, which aims to create a 3D reconstruction of the scene that can be input into the online part of the 3D ATR architecture.

1) SfM module

We propose a SfM module that exploits the gyroscope, accelerometer and visual band (RGB) camera sensor of a flying UAV platform.

Given two 2D scene images I_1, I_2 of size $m \times n$, acquired by the same camera positioned on a flying UAV at instances *t* and *t+1*, we perform keypoint detection and tracking on I_1 and I_2. Specifically, we detect and describe keypoints $p_a^{I_1}$, i.e. image pixels that are prominent among their surroundings in image I_1, by applying the good features to track (GFTT) algorithm [22] with a minimum corner quality of 10^{-3} and a 3 x 3 Gaussian filter. Then we use the Kanade-Lucas-Tomashi (KLT) tracker [23] to track these keypoints in I_2, but due to the camera's motion, only the subset $p_b^{I_1}$ $b \le a$ is tracked. For KLT, we use a forward-backward error [24] of one pixel, a 11x11 tracking window over 13 scales, and 10 iterations. Finally $p_b^{I_1}$, $p_b^{I_2}$ and the camera's transformation matrix R_{cam} at instance *t+1* in relation to *t* are input into a triangulation process to create the 3D reconstruction of the matched keypoints $p_b^{I_1}$ and $p_b^{I_2}$.

In contrast to current SfM methods that calculate R_{cam} based on the I_1, I_2 image correspondences, we calculate the camera's 6D real-world pose shift R_{cam}^* between instances *t* and *t+1* by extracting the gyroscope and the accelerometer measurements R_{cam}^t, R_{cam}^{t+1} at both instances. Specifically, we calculate:

$$R_{cam}^* = R_{cam}^{t+1} \cdot (R_{cam}^t)^{-1} \tag{1}$$

where

$$R_{cam}^t = = [R(u,v,w) \mid T(x,y,z)] \tag{2}$$

$$R(u,v,w) = \begin{pmatrix} \cos u \cdot \cos v & \cos u \cdot \sin v \cdot \sin w - \sin u \cdot \cos w & c\cos u \cdot \sin v \cdot \cos w + \sin u \cdot \sin w \\ \sin u \cdot \cos v & \sin u \cdot \sin v \cdot \sin w + \cos u \cdot \cos w & \sin u \cdot \sin v \cdot \cos w - \cos u \cdot \sin w \\ -\sin v & \cos v \cdot \sin w & \cos v \cdot \cos w \end{pmatrix} \tag{3}$$

$$T(x,y,z) = \left(\int_t^{t+1}\int_t^{t+1} \Delta a_x, \int_t^{t+1}\int_t^{t+1} \Delta a_y, \int_t^{t+1}\int_t^{t+1} \Delta a_z \right)^T \tag{4}$$

where R is the rotational and T the translational part of the transformation matrix R_{cam}^t; u, v and w are the pitch, roll and yaw, respectively; and a is the acceleration per axis on an *XYZ* reference frame set at the UAV's center of gravity. Fig. 1 shows an example of SfM 3D reconstruction. For a detailed analysis of the standard SfM method the reader is referred to [25].

We also perform SfM 3D reconstruction by exploiting the speeded up robust features (SURF) [26] keypoint detection and description technique. We apply SURF on images I_1 and I_2 to extract keypoints $p_a^{I_1}$ and $p_a^{I_2}$. SURF is applied over six scale levels with a blob threshold of 10^{-3}. The features $f_a^{I_1}$ and $f_a^{I_2}$ of $p_a^{I_1}$ and $p_a^{I_2}$ respectively, are then matched based on the nearest neighbor distance ratio (NNDR) criterion [27] with a threshold empirically set at 0.6. The correspondences $p_b^{I_1}$ and $p_b^{I_2}$ undergo the same process as described for the GFTT/ KLT case.

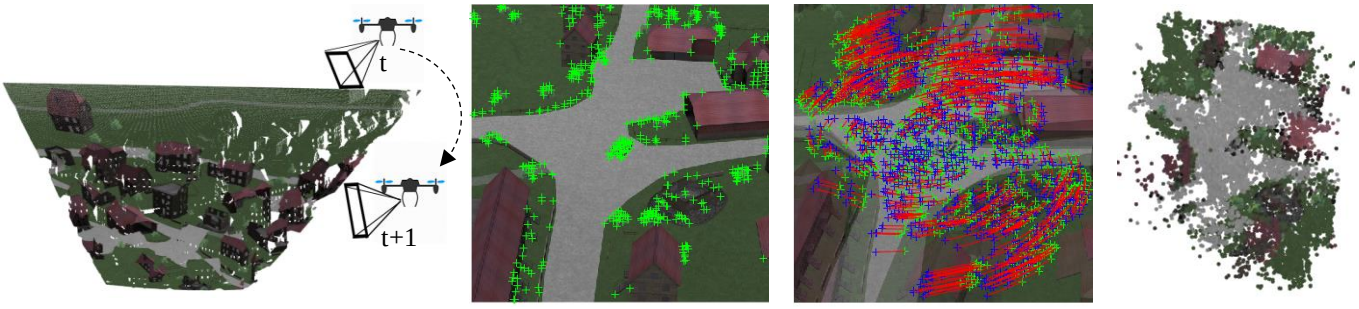

Fig. 1. SfM based concept for 3D ATR applications on UAVs (a) image pair acquisition at UAV's position *t* and *t+1* (b) keypoint detection and description (c) keypoint correspondences (d) 3D reconstruction (number of keypoints detected and matched in (b),(c) is reduced for better visualization)

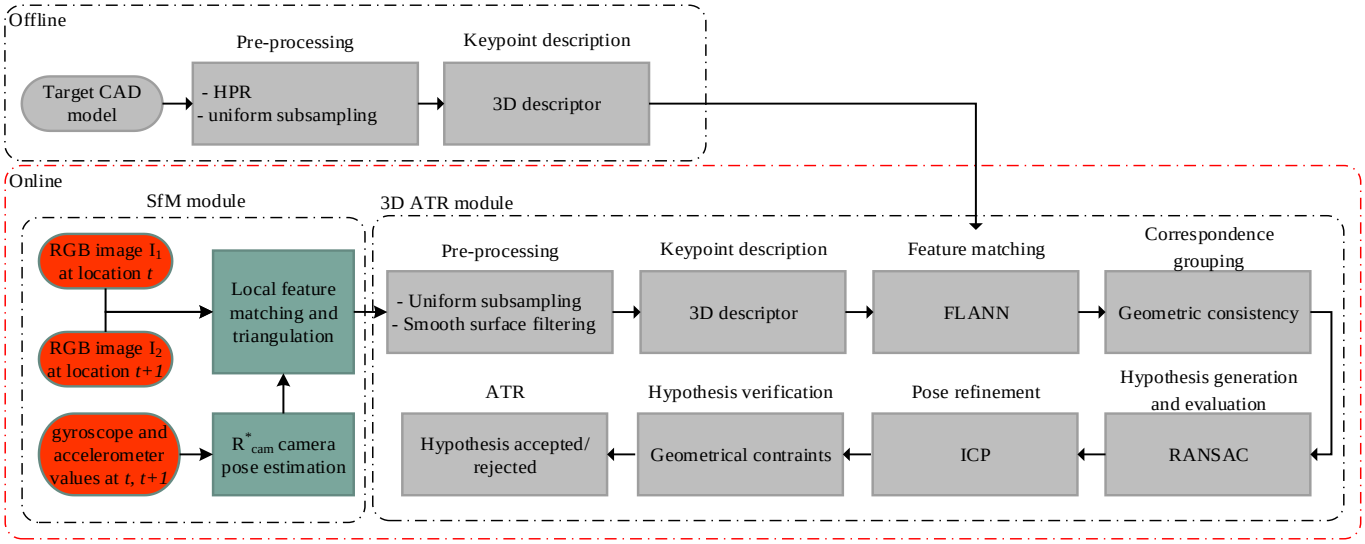

Fig. 2. SfM 3D ATR architecture

Despite the availability of several options to improve the accuracy of the point cloud reconstructed in 3D by SfM methods, these were disregarded because computational efficiency is necessary for the UAV applications considered here. Although the UAV dynamics are already known from the gyroscope and accelerometer readings and can be incorporated into the SfM estimation via a KALMAN filtering process to verify the matched keypoint correspondences, this imposes an additional processing burden and is therefore omitted. Similarly, the resulting SfM point cloud is sparse, but the additional processing cost to make it dense substantially increases the processing time, and given that the performance of the ATR is already appealing (Section III), we did not attempt to create a dense point cloud. Super-resolution [28] can improve 3D reconstruction but the resulting computational burden was too great. Finally, we did not use multiple images to construct the point cloud, allowing us to investigate the limits of SfM for 3D ATR applications.

2) 3D ATR module

During the online phase, the scene point cloud $\boldsymbol{P}$ is also uniformly subsampled at 0.3-m resolution. $\boldsymbol{P}$ is then refined to $\boldsymbol{P_f}$ by filtering its smooth surfaces based on the angular variation of the normal that is set on each vertex, compared to the normal of its surrounding vertices. Normal estimation considers fitting a plane on the six closest neighbors of the vertex for which we calculate the normal. $\boldsymbol{P_f}$ is then described using the same 3D descriptor as used for $\boldsymbol{P_m}$. Feature matching relies on a *k* nearest neighbor distance ratio (kNNDR) scheme where k=10, whereas the main keypoint matching process involves the creation of groups of $\boldsymbol{P_f}$ – $\boldsymbol{P_m}$ keypoint correspondences that are geometrically consistent. Each group $\{H_1, H_2, ..., H_g\}$, with g indicating the number of groups, is input into a random sample and consensus (RANSAC) algorithm using 1000 iterations to define a transformation hypothesis between the CAD model $\boldsymbol{P_m}$ and the scene $\boldsymbol{P_f}$. Then, each hypothesis is verified for correctness by applying it to $\boldsymbol{P_m}$ followed by alignment with $\boldsymbol{P_f}$ using an iterative closest point (ICP) scheme. Finally, the geometrical accuracy of this hypothesis is validated if the aligned model and the scene have overlapping vertices that exceed a threshold. The proposed 3D SfM ATR architecture is presented in Fig. 2.

III. EXPERIMENTS

A. Dataset

Real military datasets are restricted and we therefore used OpenFlight [29] to simulate three highly credible air-to-ground UAV reconnaissance scenarios (Table II). All scenarios considered the UAV flying a circular orbit at several UAV–target ranges, altitudes and headings, and under various

TABLE III
3D DESCRIPTORS USED

Descriptor	Descriptor Length	Implementation platform	Operating principle
SHOT	352	C++ (Matlab Exchange (MEX) wrapper)	Angular variations
USC	1980	C++ (MEX wrapper)	Accumulating points
HoD / HoD-S	240 / 40	MATLAB / MATLAB	L2-norm distances, HoD coarse and fine encryption, HoD-S coarse encryption
FPFH	33	C++ (MEX wrapper)	Angular variations
3DSC	1980	C++ (MEX wrapper)	Accumulating points
RoPS	135	MATLAB	Low order statistics

pitch, roll and yaw angles. Each scenario involved a T-72 main battle tank (MBT) in an urban environment that included clutter (non-target objects) such as buildings and trees. Depending on the UAV's flight parameters, the T-72 target might be partially or even completely occluded by clutter. Notably, our scenarios simulated not only the size of the target, which depends on the UAV–target range, but for the LIDAR case they also considered the laser spot size and how this affects the LIDAR point cloud. In contrast to previous studies [7,30,31], our military scenarios were affected by more parameters and are therefore more challenging and realistic. For each scene, we generated a 3D LIDAR point cloud and the corresponding 2D visual image. Camera intrinsic and extrinsic parameters are the ones used while creating the scenarios.

B. Experimental setup

We evaluated the effectiveness of the new SfM-based ATR using a multi-level scheme, i.e. challenging the effectiveness of several current 3D ATR descriptors on SfM point clouds compared to LIDAR point clouds as well as classic 2D ATR methods based on local features.

Specifically, for the SfM-based 3D ATR, we exploited the ATR pipeline presented in Fig. 2, but for the LIDAR 3D ATR we replaced the SfM module with the LIDAR-based point cloud. In both cases, we evaluated the following descriptors: signature of histograms of orientations (SHOT) [32], rotational projections statistics (RoPS) [33], fast point feature histograms (FPFH) [34], 3D shape context (3DSC) [35], unique shape context (USC) [36], histogram of distances (HoD) [37] and histogram of distances – short (HoD-S) [10]. The description radius of each 3D descriptor was $\rho \cdot r$, where r is the average point cloud resolution of the CAD model [32,33,38] and ρ a multiplier as suggested by the authors of each descriptor (e.g. for HoD and HoD-S, r is the scene resolution [37]). Table III presents each 3D descriptor and its parameters, which were fixed either to those originally proposed by their authors or to their point cloud library implementation [37,39]. Given that each 3D descriptor was applied on a spherical volume $\boldsymbol{V}$ of radius ρ centered at a keypoint p, the operating principle of each 3D descriptor can be summarized as follows:

a. SHOT [32] establishes a local reference frame (LRF) on p and divides $\boldsymbol{V}$ into a number of sub-volumes along the azimuth, the elevation and the radius. For each sub-volume, SHOT encodes the normal variation among p (including its neighbouring vertices) with the normal of each sub-volume.

b. RoPS [33] establishes a LRF on p, then $\boldsymbol{V}$ is rotated around each axis of the LRF's coordinate frame and is finally projected on each of the coordinate planes. RoPS encryption involves a low-order moment and entropy description of each projection, and these are concatenated to formulate a histogram.

c. FPFH [34] establishes a LRF on p, and for each vertex belonging to $\boldsymbol{V}$, FPFH encodes the angular relationship between p and its neighbours as provided by the LRF. Finally, that angular relationship is transformed into a histogram.

d. For 3DSC [35], a local reference axis (LRA) is established on p, aligned to the normal produced by the vertices in $\boldsymbol{V}$, and $\boldsymbol{V}$ is divided into a number of sub-volumes along the azimuth, elevation, and radial dimension. The 3DSC descriptor is established by accumulating a weighted sum of the points within each sub-volume. Weights are proportional to the sub-volume to centre-of-V distance. 3DSC is LRA-based and compensates for 360° azimuthal rotation by describing $\boldsymbol{V}$ in multiple azimuthal orientations. USC [36] is identical to 3DSC but the LRA is replaced with an LRF.

e. HoD [37] calculates the point-pair L2-norm distance distributions of the vertices within $\boldsymbol{V}$. L2-distances are encoded in a coarse and a fine manner. HoD-S [10] involves only the coarse component of HoD.

In addition to the 3D SfM *vs* 3D LIDAR comparison, we also compared 3D SfM against classic 2D local feature ATR. For that purpose, we used the pipeline presented in Fig. 3 with the 2D keypoint descriptors and detectors as shown in Table IV. This table also presents the parameters used for each keypoint detector and descriptor combination to maximize its ATR performance.

The parameters of each remaining combination were fixed to those originally proposed by the author. Feature matching was based on the NNDR criterion [27] with a threshold of 0.8 and the M-estimator sample consensus (MSAC) algorithm [40] was used to refine the correspondences.

TABLE II
SCENARIO PARAMETERS

Scenario N°	1	2	3
N° of runs	4	4	1
Obliquity (°)	0°–45° per 15°	0°–45° per 15°	30°
UAV-target (m)	50	100	200
Resolution (cm)	11	18	30
Scenes with target/out of total	334/345	327/364	78/78

TABLE IV
2D KEYPOINT DETECTION AND DESCRIPTION COMBINATION USED

ID	Keypoint detector	Keypoint descriptor	Descriptor length	Implementation platform	Tuned parameters
#1	GFTT	Fast Retina Keypoint (FREAK)	64	C++ (MEX wrapper)	Min corner quality 10^{-3} Gaussian filter size 3x3
#2	SURF	SURF	64	C++ (MEX wrapper)	Scale levels 6
#3	Features from Accelerated Segment Test (FAST)	Binary Robust Invariant Scalable Keypoints (BRISK)	64	C++ (MEX wrapper)	Min contrast 10^{-3}
#4	FAST	FREAK	64	C++ (MEX wrapper)	Min corner quality 10^{-3} and Min contrast 10^{-3}

C. Performance metric

ATR performance was evaluated using the F1 score [10]:

$$F1-score = \frac{2\#TP}{2\#TP + \#FP + \#FN} \quad (5)$$

where # denotes the number of the metric that follows, i.e. true positive (*TP*), false positive (*FP*) and false negative (*FN*). We selected the *F1-score* metric because it encapsulates the classic precision and recall metrics without involving the true negative (*TN*) metric. This is important because in a number of runs per scenario the target is always present, i.e. *TN*=0, and thus $recall = \#TP \cdot (\#TP + \#TN)^{-1}$ would be biased.

As previously reported [10], these metrics not only compare the ATR prediction state with the actual state but also consider the Euclidean distance-based translational error T_{error} between the ground truth position of the target in the scene and its estimated final position. Hence, for a *TP* match the algorithm provides a transformation hypothesis for a scene where a target is present and T_{error}<2m. For an *FP* match, the algorithm provides a hypothesis for a scene that does not have a target or has a target with T_{error}>2m. This dual constraint, i.e. target presence in the scene and target localization accuracy (T_{error}), ensures that the *FP* match metric is not biased for scenarios in which the target is always present. Finally, the *FN* match case occurs if the algorithm does not provide a hypothesis for a scene that has a target. For fairness, T_{error} was also extended to facilitate the 2D ATR scheme.

D. Assessment

We evaluated the ATR performance in terms of UAV–target range, obliquity variation, processing time, template storage, descriptor compactness, robustness to shot noise, and to Gaussian noise. The trials involved a UAV reconnaissance application for which we reduced the processing time of the 3D ATR by exploiting a single CAD model, whereas for the 2D ATR we minimised the number of templates as suggested [1]. Hence, we used 12 images of the target, evenly spaced across the 0–360° azimuthal viewing angle, and these images were cropped from the first trial of the first scenario. It is worth noting that since the templates are cropped from the evaluation scenes, the performance of 2D ATR is positively biased. To balance this we only exploit image templates from a single scenario and run, while the experiments involve nine runs in total (Table II). Trials are implemented on an i7 at 2.6GHz with 16GB RAM.

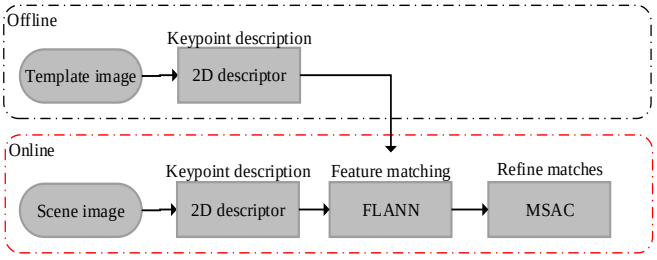

Fig. 3. 2D ATR architecture used for comparative purposes

1) UAV-target range evaluation

In this trial, we compared the performance of the 3D LIDAR, 3D SfM and 2D ATR in relation to the UAV–target distance. Fig. 4a shows that the LIDAR and SfM 3D ATR performed equally well at 50 m UAV–target range, and outperformed 2D ATR because the SfM point cloud preferentially reconstructs the central region of the image close to the target, reducing misclassifications. We found that GFTT has a small performance advantage over SURF SfM that was consistent among all 3D descriptors. Furthermore, GFTT outperformed the 2D competitors, but was still inferior to both the LIDAR and SfM 3D ATR techniques.

When the UAV–target distance increased to 100 m (Fig. 4b), the performance of all three solutions declined. When the distance increased to 200 m (Fig. 4c), the 3D SfM achieved the best performance and the 2D ATR the worst, as explained in more detail below.

SfM is created by matching 2D keypoints from two images at the same range at 100 or 200 m. Hence, 2D features are detected within the same scale and can be matched in sequential images for 3D SfM reconstruction. Because the UAV flies a circular orbit, 3D reconstruction is more accurate closer to the center of the orbit. In contrast, 2D ATR encodes keypoints from a template presenting the MBT at a range of 50 m and aims to match these keypoints with those detected on a MBT at a different scale. Especially for the 200 m range, the MBT in the scene is four times further away than its template. That scale difference exceeds the scale invariance of all 2D descriptors. In addition, templates are derived from 30° obliquity whereas the angles are evaluated in the range 0–45°, exceeding the out-of-plane invariance of the 2D descriptors. Even though these are acknowledged as problems in 2D ATR, we intentionally adopted a small template [1] to demonstrate the advantage of SfM 3D ATR under a single-template scheme. Increasing the 2D templates to accommodate several target poses and scales affects the computational and storage requirements, which are not always affordable, especially for time-critical applications. An analysis of the processing time and storage requirements is presented in Section III-D-3.

Unsurprisingly the performance of 3D LIDAR ATR declined at a range of 200 m because the laser spot size increases as the beam propagates through the atmosphere,

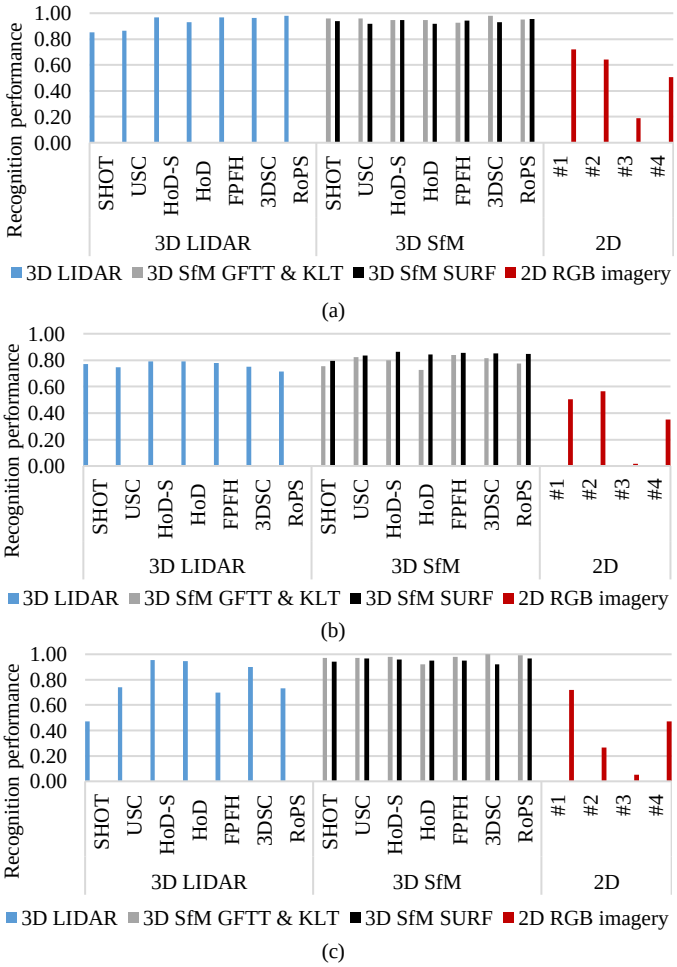

Fig. 4. ATR for 3D LIDAR *vs* 3D SfM *vs* 2D ATR in relation to UAV–target range. (a) Scenario 1 – 50 m. (b) Scenario 2 – 100 m. (c) Scenario 3 – 200 m

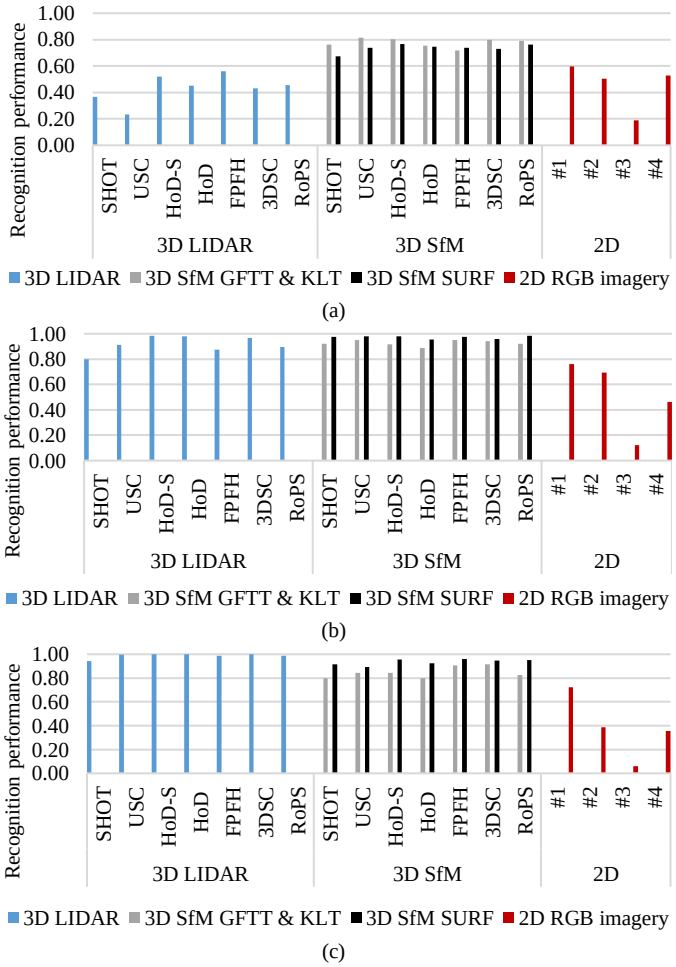

Fig. 5. ATR for 3D LIDAR *vs* 3D SfM *vs* 2D ATR in relation to target obliquity over all scenarios. (a) Low – 15°. (b) Medium – 30°. (c) High – 45°

forcing the MBT in the scene to have simultaneously a smaller size and a lower resolution.

2) UAV-target obliquity evaluation

This trial evaluated robustness in terms of obliquity variation but still considered the three UAV–target ranges. Even though the trials considered obliquity values of 0–45° in increments of 15°, to improve clarity we focus on the ATR performance for low, medium and large obliquity angles of 0°, 30° and 45°, respectively (Fig. 5).

For the low-angle test, 3D SfM achieved the highest ATR performance by a large margin, with recognition rates of 81.5% for the GFTT with USC, and 76.7% for the SURF with HoD-S. The maximum performance of 3D LIDAR was 56% with FPFH, whereas 2D ATR achieved only 60%recognition. For the medium-angle test, 3D SfM and 3D LIDAR performed equally well at all three UAV–target ranges, achieving scores of 98% and 99%, respectively. Although 2D ATR fared better than in the low-angle test, it was still inferior to the 3D solutions, with a 76% recognition rate. The 3D LIDAR ATR gained near-perfect scores in the high-angle test, and SURF SfM ATR was only mildly less successful, achieving a 96% recognition rate. Furthermore, GFTT SfM ATR and 2D ATR achieved scores of 90.8% and 72.3%, respectively. The ATR performance attained is explained below.

For the low-angle test, 3D LIDAR suffered from a high rate of *FP* matches, leading to a low F1-score, because LIDAR encapsulates a greater part of the scene. In contrast, in the context of 3D SfM, the further away a keypoint is from the camera's optical axis, the larger its frame-to-frame motion. If this motion exceeds the one-pixel threshold, it is not reconstructed. Therefore, the 3D SfM favors 3D reconstruction near the camera optical axis and thus achieves a better performance than the 3D LIDAR point cloud. Even if the two images used for SfM lack a MBT close to the camera's optical axis (such that the target is not reconstructed in 3D), the MBT will occupy the center of subsequent images as the UAV moves and thus the target will be reconstructed at some point. The 2D ATR did not perform well because both the distance (scale) and obliquity exceeded the invariance of the 2D descriptors. For the medium and high angle tests, more of the MBT's top-view was revealed, which is more distinctive than the side-view, favoring ATR. The overall performance of each method is shown in Fig. 6, highlighting the better performance of SfM 3D ATR compared to 3D LIDAR ATR. Fig. 7 shows 3D ATR examples for both methods.

3) Computational and storage requirement analysis

Recognition performance and computational efficiency are equally important for an ATR system. We therefore compared the 3D SfM, 3D LIDAR and 2D ATR methods in terms of their processing burden (Table V). Although SfM requires the scene to be reconstructed in 3D before activating the rest

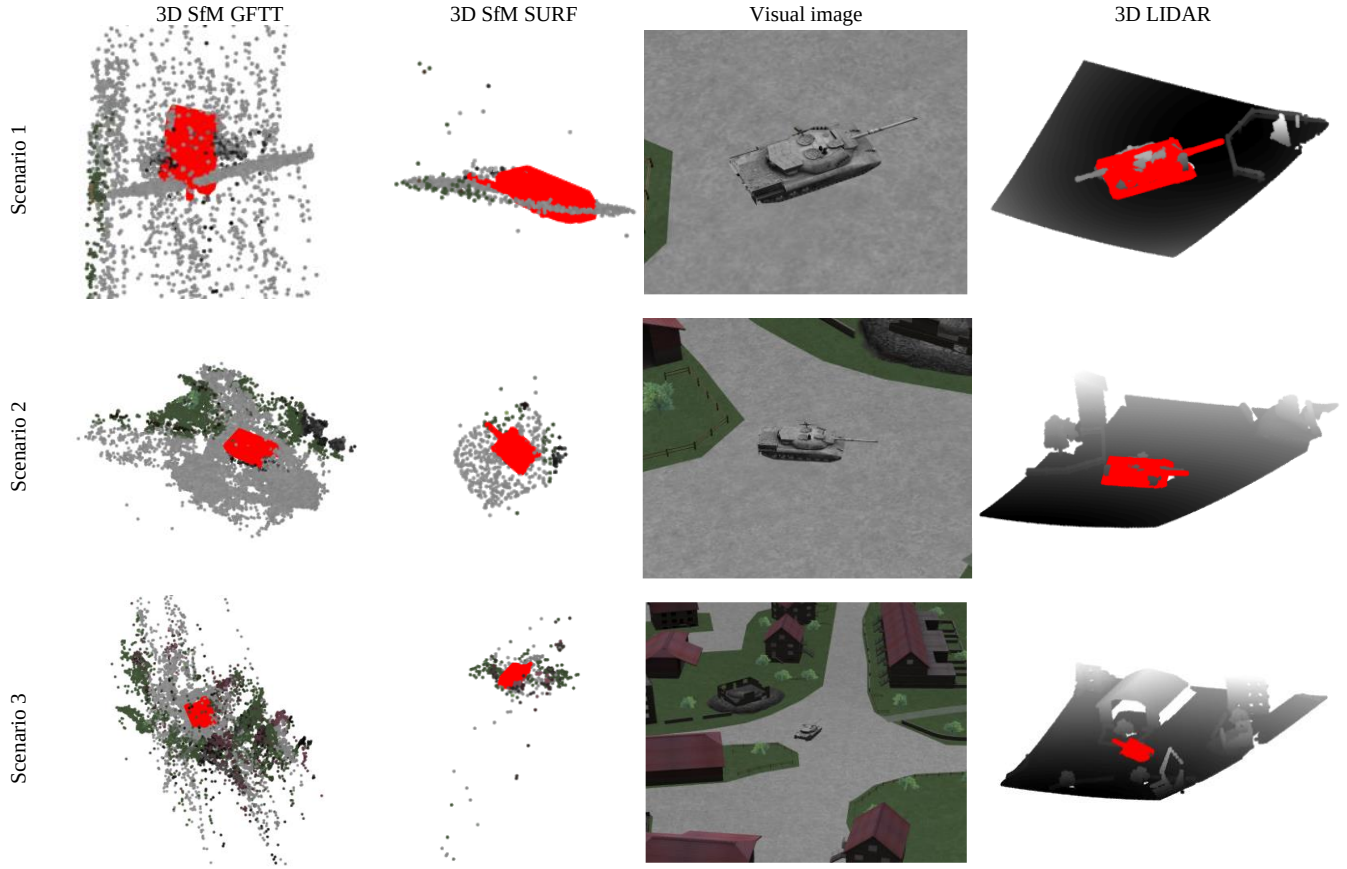

Fig. 7. Examples of 3D ATR with SfM, exploiting only two images from the visual domain

of the pipeline (Fig. 2), 3D SfM is faster than 3D LIDAR because the SfM-based point cloud is sparser, speeding up the entire recognition process. Indeed, GFTT SfM produces a point cloud in the order of 10,000 vertices, whereas the equivalent values for SURF SfM and LIDAR are 500 and 260,000, respectively. A 3D SfM exploiting GFTT keypoints combined with the HoD-S descriptor therefore requires only 0.25 s for completion, whereas the less efficient 3DSC needs 0.69 s and 3D SfM with SURF features needs up to 0.84 s. In contrast, the fastest 3D LIDAR ATR was based on HoD-S (1.6 s) and the least efficient was RoPS (14.3 s). It is evident that the processing efficiency of the proposed 3D SfM architecture is at least one order of magnitude faster than 3D LIDAR ATR.

A detailed processing breakdown is shown in Fig. 8, indicating that the 3D description of the SfM point cloud vertices is almost eight times faster than the LIDAR-based point cloud due to the sparsity of the SfM point cloud. This

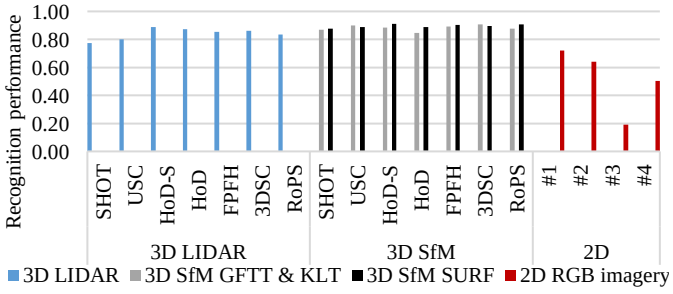

Fig. 6. Overall ATR performance of 3D LIDAR *vs* 3D SfM *vs* 2D

advantage is also evident from the considerably faster keypoint matching, correspondence hypothesis evaluation and verification achieved by both SfM methods.

As expected, the shortest processing time was observed for 2D ATR. Although this is an appealing property, the template is reduced to a minimum, so expansion to provide more instances of the target at various viewing angles and ranges would increase the overall processing time. Furthermore, in terms of the storage capacity needed for template features, even in this minimal template case, the 2D solutions already have greater requirements than their 3D counterparts because the 3D template is subsampled and only a few vertices from the entire CAD model are encoded.

4) Matching accuracy

We also validated the 3D SfM concept by highlighting the 3D translational error (T_{error}) of each descriptor. Fig. 9 shows the T_{error} of the three 3D approaches (GFTT SfM, SURF SfM and LIDAR) for the UAV–target range of 200 m at 30° obliquity. For greater clarity, we evaluated the matching accuracy for the third scenario alone, which involves the largest UAV–target range among the three scenarios, and therefore is the most challenging.

As anticipated, 3D LIDAR generated the smallest errors because the target within the point cloud was more complete than its corresponding sparse SfM reconstructions. Even so, both SfM solutions still produced low T_{error} values, confirming that the suggested SfM ATR architecture is an appealing creates that focuses on the target. For the GFTT SfM method, the largest T_{error} was generated by HoD-S (0.51 m),

TABLE V
REQUIREMENT ANALYSIS

		3D							2D			
		SHOT	USC	HoD-S	HoD	FPFH	3DSC	RoPS	#1	#2	#3	#4
Template storage (KB)	SfM / LIDAR	2.9	15.9	0.32	1.92	0.26	15.9	1.1	2213	1389	1224	3106
Processing time / scene (s)	SfM GFTT	**0.42**	**0.57**	**0.25**	**0.26**	**0.38**	**0.69**	**0.30**				
	SfM SURF	0.73	0.78	0.54	0.55	0.66	0.84	0.62	0.58	0.05	0.09	0.74
	LIDAR	4	11.5	1.6	2.2	2.5	11.1	14.3				

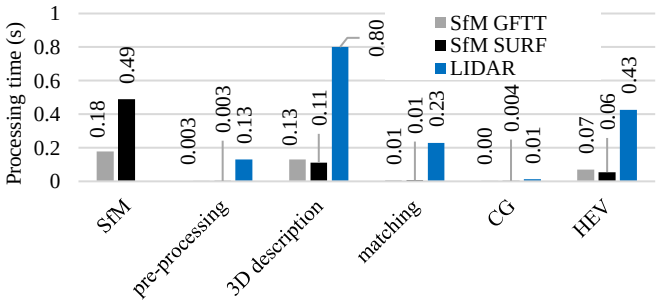

Fig. 8. Processing time breakdown (CG: correspondence grouping, HEV: hypothesis evaluation and verification)

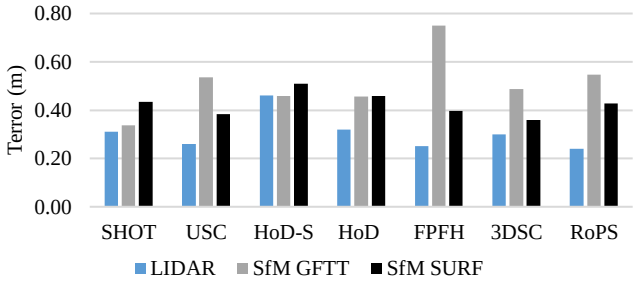

Fig. 9. Translational error evaluation

and for SURF SfM the largest value was generated by FPFH (0.75 m), but all these values are still very low. T_{error} fluctuations among the descriptors are related to the sparsity of the point cloud, whether the 3D descriptor employs an LRF/LRA or not, and the concept used to estimate the LRF or LRA.

5) Compactness

This metric indicates the description power per element of a descriptor [10]:

$$compactness = \frac{F1-score}{\#descriptor\ cardinality} \tag{6}$$

Fig. 10 shows that for both LIDAR and SfM, HoD-S and FPFH were the most compact, with 3D GFTT-based SfM displaying a minor advantage. The greater compactness of HoD-S and FPFH reflect the small feature length/cardinality of these descriptors, which in parallel achieve a competitive ATR performance. The least compact were USC and 3DSC, because despite achieving better ATR performance compared to FPFH, their large feature length severely compromised their compactness.

Regarding the 2D descriptors, even though their feature length is small, they all have a small compactness value due to their relatively poor ATR performance.

6) Robustness to Shot noise

We compared the robustness of the proposed and competing ATR methods against shot noise by modeling shot noise with a Poisson distribution. Shot noise was applied on the core data required by each method. Hence, for the SfM 3D ATR and the 2D ATR tests, we applied shot noise directly to the 2D RGB imagery, whereas for the 3D LIDAR ATR test we applied shot noise to the vertices of the point cloud.

Specifically, we independently manipulated each pixel of the 2D scene image $I_1(i,j)$, $1 \le i \le m$ and $1 \le j \le n$ according to:

$$I_1(i,j) = \mathrm{e}^{-I_1(i,j)} \frac{I_1(i,j)}{k!} \tag{7}$$

where $k \in \mathbb{N}^+$ randomly chosen. In the same manner, we applied shot noise to I_2. For the LIDAR 3D ATR test, we independently manipulated the z-coordinate of each vertex in the LIDAR point cloud $\boldsymbol{P}$ according to the corresponding depth value of the 2D depth image D that the LIDAR creates:

$$\boldsymbol{P}(x,y,z) = \left[x \quad y \quad \mathrm{e}^{-D(ii,jj)} \frac{D(ii,jj)}{k!} \right] \tag{8}$$

where *ii* and *jj* are the pixel coordinates of *D*.

Fig. 11 clearly shows that the SfM 3D ATR architecture outperforms both competitors regardless of the descriptor. This is important because it demonstrates the advantages of using SfM rather than LIDAR 3D data. The robustness of SfM 3D ATR reflects the robustness of the 2D local feature methods used in our SfM module, which successfully matched the images (corrupted by shot noise) acquired from the UAV's camera in order to create an accurate 3D scene representation. As expected, the performance of the 2D ATR pipeline was poor for the reasons presented in Sections III-D-1 and III-D-2.

7) Robustness to Gaussian noise

We also evaluated the robustness of the proposed ATR technique under $\sigma=\{10,30\}cm$ Gaussian noise levels [10]. Similarly to the shot noise trial, we applied noise directly to the 2D RGB imagery for both 3D SfM and 2D ATR, whereas for the LIDAR 3D ATR the Gaussian noise was applied to the vertices of the point cloud.

Fig. 12a shows that for the σ = 10 cm Gaussian noise test, 3D SfM ATR achieved a more stable performance, which was less dependent on the descriptor. In contrast, even though 3D LIDAR ATR combined with the HoD and HoD-S descriptors achieved the highest ATR performance, our trials demonstrate that the selected descriptor had a substantial impact on the ATR performance.

For the σ = 30 cm Gaussian noise test, 3D SfM ATR achieved a higher overall ATR capability (Fig. 12b). This was more evident for the GFTT and KLT combination, where the

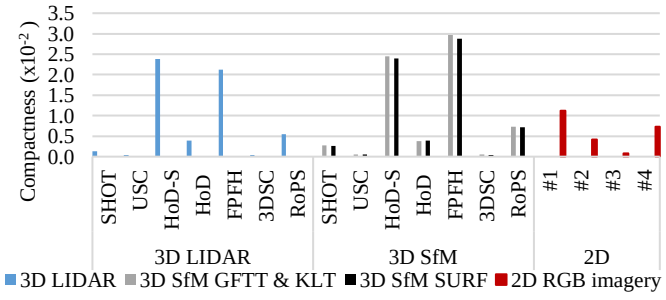

Fig. 10. Compactness

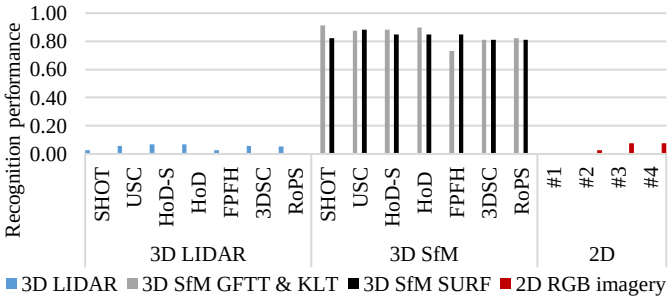

Fig. 11. Robustness to shot noise

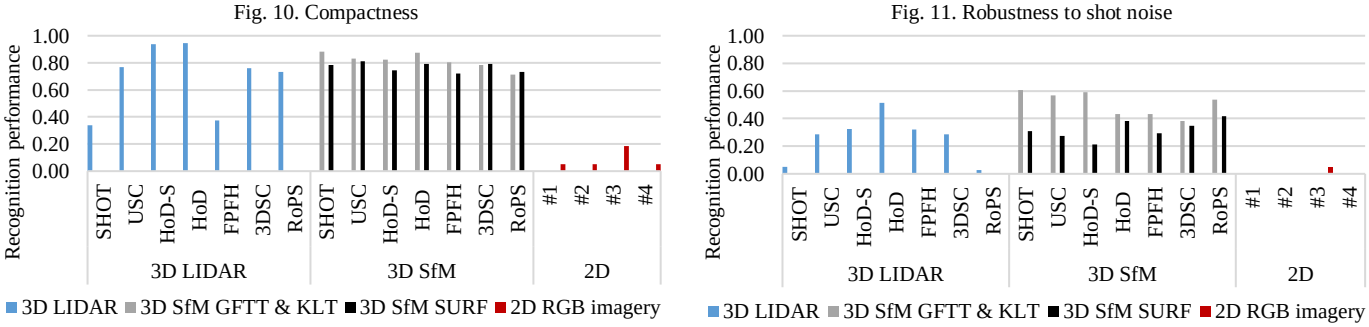

Fig. 12. Robustness to Gaussian noise with zero mean and (a) σ = 10 cm (b) σ = 30 cm

majority of the descriptors achieved higher recognition rates than the best-performing 3D LIDAR descriptor.

IV. Conclusion

We have developed a passive 3D ATR architecture appropriate for small, low-cost UAV platforms. Our architecture exploits the UAV's onboard sensors, i.e. visual band camera, gyroscope and accelerometer, in order to create passive 3D reconstructions of the UAV's surroundings. The 3D scene thus created is input into a 3D ATR pipeline. The method is appealing because it combines the advantages of 3D and 2D object recognition. Specifically, it combines the advantages of 3D object recognition, such as pose and illumination invariance, exploiting the underlying structure of the target and reducing the template size to a single 3D CAD model. In addition, it also preserves the advantages of 2D object recognition, resulting in a small processing burden, low hardware costs (camera *vs* LIDAR), faster data acquisition, longer operating range, and undetectable passive operation.

We evaluated the new SfM ATR scheme by exploiting two 2D keypoint detection and description techniques, i.e. the GFTT with a KLT tracker and the SURF with an NNDR criterion, and we tested these against classic 3D LIDAR ATR and 2D visual ATR. We measured target recognition performance over several UAV–target ranges and obliquities, as well as evaluating processing efficiency, translational matching accuracy, robustness to shot noise and to Gaussian noise, confirming its appealing features. One limitation of our technique compared to LIDAR 3D is the constraint of sufficient lighting conditions, which reflect the camera's limitations. However, in the future we intend to extend the 3D SfM ATR concept to operate on low-light visual band cameras in order to improve the usability of the suggested architecture to include extreme lighting scenarios.

References

[1] G.J. Gray, N. Aouf, M.A. Richardson, B. Butters, R. Walmsley, E. Nicholls, Feature-based recognition approaches for infrared anti-ship missile seekers, Imaging Sci. J. 60 (2012) 305–320. doi:10.1179/1743131X12Y.0000000012.

[2] O. Kechagias-Stamatis, N. Aouf, D. Nam, Multi-Modal Automatic Target Recognition for Anti-Ship Missiles with Imaging Infrared Capabilities, in: 2017 Sens. Signal Process. Def. Conf., IEEE, 2017: pp. 1–5. doi:10.1109/SSPD.2017.8233244.

[3] R. Paladini, M. Martorella, F. Berizzi, Classification of Man-Made Targets via Invariant Coherency-Matrix Eigenvector Decomposition of Polarimetric SAR/ISAR Images, IEEE Trans. Geosci. Remote Sens. 49 (2011) 3022–3034. doi:10.1109/TGRS.2011.2116121.

[4] D. Perissin, A. Ferretti, Urban-Target Recognition by Means of Repeated Spaceborne SAR Images, IEEE Trans. Geosci. Remote Sens. 45 (2007) 4043–4058. doi:10.1109/TGRS.2007.906092.

[5] M. Martorella, E. Giusti, A. Capria, F. Berizzi, B. Bates, Automatic Target Recognition by Means of Polarimetric ISAR Images and Neural Networks, IEEE Trans. Geosci. Remote Sens. 47 (2009) 3786–3794. doi:10.1109/TGRS.2009.2025371.

[6] M. Prashnani, R.S. Chekuri, Identification of military vehicles in hyper spectral imagery through spatio-spectral filtering, in: 2013 IEEE Second Int. Conf. Image Inf. Process., IEEE, 2013: pp. 527–532. doi:10.1109/ICIIP.2013.6707648.

[7] A. Vasile, R. Marino, Pose-independent automatic target detection and recognition using 3D laser radar imagery, Lincoln Lab. J. 15 (2005) 61–78.

[8] C. Grönwall, Ground object recognition using laser radar data: geometric fitting, performance analysis, and applications, Linköping, Sweden, 2006.

[9] O. Kechagias-Stamatis, N. Aouf, M.A. Richardson, 3D automatic target recognition for future LIDAR missiles, IEEE Trans. Aerosp. Electron. Syst. 52 (2016) 2662–2675. doi:10.1109/TAES.2016.150300.

[10] O. Kechagias-Stamatis, N. Aouf, Evaluating 3D local descriptors for future LIDAR missiles with automatic target recognition capabilities, Imaging Sci. J. 65 (2017) 428–437. doi:10.1080/13682199.2017.1361665.

[11] O. Kechagias-Stamatis, N. Aouf, G. Gray, L. Chermak, M. Richardson, F. Oudyi, Local feature based automatic target recognition for future 3D active homing seeker missiles, Aerosp. Sci. Technol. 73 (2018) 309–317. doi:10.1016/j.ast.2017.12.011.

[12] S. Matteoli, G. Corsini, M. Diani, G. Cecchi, G. Toci, Automated Underwater Object Recognition by Means of Fluorescence LIDAR, IEEE Trans. Geosci. Remote Sens. 53 (2015) 375–393.

doi:10.1109/TGRS.2014.2322676.
[13] S.K. Rogers, J.M. Colombi, C.E. Martin, J.C. Gainey, K.H. Fielding, T.J. Burns, D.W. Ruck, M. Kabrisky, M. Oxley, Neural networks for automatic target recognition, Neural Networks. 8 (1995) 1153–1184.
[14] A.S. Mian, M. Bennamoun, R. Owens, Three-Dimensional Model-Based Object Recognition and Segmentation in Cluttered Scenes, IEEE Trans. Pattern Anal. Mach. Intell. 28 (2006) 1584–1601. doi:10.1109/TPAMI.2006.213.
[15] S.Y. Bao, S. Savarese, Semantic structure from motion: a novel framework for joint object recognition and 3d reconstruction, in: Outdoor Large-Scale Real-World Scene Anal., Springer, 2012: pp. 376–397.
[16] G.J. Brostow, J. Shotton, J. Fauqueur, R. Cipolla, Segmentation and recognition using structure from motion point clouds, in: Eur. Conf. Comput. Vis., 2008: pp. 44–57.
[17] B. Leibe, N. Cornelis, K. Cornelis, L. Van Gool, Dynamic 3d scene analysis from a moving vehicle, in: Comput. Vis. Pattern Recognition, 2007. CVPR'07. IEEE Conf., 2007: pp. 1–8.
[18] M. Shim, S. Yilma, K. Bonner, A robust real-time structure from motion for situational awareness and RSTA, (2008) 696205. doi:10.1117/12.778074.
[19] Y. Wei, L. Kang, B. Yang, L. Wu, Applications of structure from motion: a survey, J. Zhejiang Univ. Sci. C. 14 (2013) 486–494. doi:10.1631/jzus.CIDE1302.
[20] Y.-P. Huang, L. Sithole, T.-T. Lee, Structure From Motion Technique for Scene Detection Using Autonomous Drone Navigation, IEEE Trans. Syst. Man, Cybern. Syst. (2017) 1–12. doi:10.1109/TSMC.2017.2745419.
[21] S. Katz, A. Tal, R. Basri, Direct visibility of point sets, ACM Trans. Graph. 26 (2007) 24. doi:10.1145/1276377.1276407.
[22] Jianbo Shi, Tomasi, Good features to track, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. CVPR-94, IEEE Comput. Soc. Press, 1994: pp. 593–600. doi:10.1109/CVPR.1994.323794.
[23] B.D. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, Proc. 7th Int. Jt. Conf. Artif. Intell. - Vol. 2. (1981) 674–679. http://dl.acm.org/citation.cfm?id=1623280 (accessed March 15, 2017).
[24] Z. Kalal, K. Mikolajczyk, J. Matas, C. Republic, Forward-Backward Error : Autonomous Identification of Tracking Failures, (2010) 23–26. doi:10.1109/ICPR.2010.675.
[25] R. Cipolla, Structure From Motion, Univ. Cambridge Lect. Notes. (2008) 1–49. http://academic.research.microsoft.com/Search?query=structure+from+motion.
[26] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-Up Robust Features (SURF), Comput. Vis. Image Underst. 110 (2008) 346–359. doi:10.1016/j.cviu.2007.09.014.
[27] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, 2003 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognition, 2003. Proceedings. 2 (n.d.) II-257-II-263. doi:10.1109/CVPR.2003.1211478.
[28] S. Knorr, M. Kunter, T. Sikora, Stereoscopic 3D from 2D video with super-resolution capability, Signal Process. Image Commun. 23 (2008) 665–676.
[29] Presagis, OpenFlight visual simulation, (n.d.). http://www.presagis.com/products_services/standards/openflight/ (accessed August 1, 2016).
[30] X.L. Xiaofeng Li, J.X. Jun Xu, J.L. Jijun Luo, L.C. Lijia Cao, S.Z. Shengxiu Zhang, Ground target recognition based on imaging LADAR point cloud data, Chinese Opt. Lett. 10 (2012) S11002-311005. doi:10.3788/COL201210.S11002.
[31] C. Gronwall, Ground target recognition using rectangle estimation, IEEE Trans. Image Process. 15 (2006) 3400–3408.
[32] S. Salti, F. Tombari, L. Di Stefano, SHOT: Unique signatures of histograms for surface and texture description, Comput. Vis. Image Underst. 125 (2014) 251–264. doi:10.1016/j.cviu.2014.04.011.
[33] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, J. Wan, Rotational Projection Statistics for 3D Local Surface Description and Object Recognition, Int. J. Comput. Vis. 105 (2013) 63–86. doi:10.1007/s11263-013-0627-y.
[34] R.B. Rusu, N. Blodow, M. Beetz, Fast Point Feature Histograms (FPFH) for 3D registration, in: 2009 IEEE Int. Conf. Robot. Autom., IEEE, 2009: pp. 3212–3217. doi:10.1109/ROBOT.2009.5152473.
[35] A. Frome, D. Huber, R. Kolluri, T. Bülow, J. Malik, Recognizing Objects in Range Data Using Regional Point Descriptors, in: ECCV, 2004: pp. 224–237. doi:10.1007/978-3-540-24672-5_18.
[36] F. Tombari, S. Salti, L. Di Stefano, Unique shape context for 3d data description, in: Proc. ACM Work. 3D Object Retr. - 3DOR '10, ACM Press, New York, New York, USA, 2010: p. 57. doi:10.1145/1877808.1877821.
[37] O. Kechagias-Stamatis, N. Aouf, Histogram of distances for local surface description, in: 2016 IEEE Int. Conf. Robot. Autom., IEEE, 2016: pp. 2487–2493. doi:10.1109/ICRA.2016.7487402.
[38] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, J. Wan, TriSI : A Distinctive Local Surface Descriptor for 3D Modeling and Object Recognition, in: 8th Int. Conf. Comput. Graph. Theory Appl., Barcelona, Spain, 2013. doi:10.5220.
[39] Y. Guo, M. Bennamoun, F. Sohel, M. Lu, J. Wan, N.M. Kwok, A Comprehensive Performance Evaluation of 3D Local Feature Descriptors, Int. J. Comput. Vis. 116 (2016) 66–89. doi:10.1007/s11263-015-0824-y.
[40] P.H.S. Torr, A. Zisserman, MLESAC: A new robust estimator with application to estimating image geometry, Comput. Vis. Image Underst. 78 (2000) 138–156.

Odysseas Kechagias-Stamatis received the MSc degree in Guided Weapon Systems and the PhD degree in 3D ATR for missile platforms from Cranfield University, U.K. in 2011 and 2017 respectively. His research interests include 2D/3D object recognition and tracking, data fusion and autonomy of systems.

Professor Nabil Aouf is currently Head of the Signals and Autonomy group, Centre for Electronic Warfare information and Cyber, Cranfield University, U.K. He has authored over 100 publications in high caliber in his domains of interest. His research interests are aerospace and defense systems, information fusion and vision systems, guidance and navigation, tracking, and control and autonomy of systems. He is an Associate Editor of the Imaging Science Journal